

УДК 519.854.2



Ф.Г. Ващук¹, О.О. Мельник²

¹ ЗакДУ, м. Ужгород, Україна, ZakDU@ukrport.net;

² ЗакДУ, м. Ужгород, Україна, me3000@ukr.net

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ АЛГОРИТМУ СКЛАДАННЯ РОЗКЛАДІВ СУМАРНОГО ВИПЕРЕДЖЕННЯ І ЗАПІЗНЕННЯ ІЗ НАЛАГОДЖЕННЯМИ, ЩО ЗАЛЕЖАТЬ ВІД ПОСЛІДОВНОСТІ

Розглядається задача складання розкладів за критерієм мінімізації сумарного випередження і запізнення відносно директивних строків при виконанні незалежних завдань одним приладом при наявності налагоджень (МВЗН). Проведені експериментальні дослідження запропонованого евристичного алгоритму пошуку локального оптимального розв'язку задачі показали, що він генерує за прийнятний час розв'язки, достатньо близькі до оптимальних, та може ефективно застосовуватись для розв'язання задач великої розмірності.

СКЛАДАННЯ РОЗКЛАДІВ, КРИТЕРІЙ МІНІМІЗАЦІЇ, НАЛАГОДЖЕННЯ, ФАКТОР ЗАПІЗНЕННЯ

Вступ

У [1, 2] описано новий підхід до розв'язання задачі за критерієм мінімізації сумарного випередження і запізнення (МВЗ). У статті [3] розширюється цей підхід для більш загальної та більш практичної задачі, що виникає особливо у виробничих задачах складання розкладів – задачі із часами налагодження, – та розробляється ефективний алгоритм A її розв'язання. Стаття присвячена дослідженню ефективності алгоритму A . Оцінювалась ефективність алгоритму у порівнянні з розробленим алгоритмом гілок і границь, вплив розмірності задачі на час її розв'язання, вплив діапазону директивних строків R і фактору запізнення T .

1. Постановка задачі

Множина з n незалежних завдань $J = \{j_1, j_2, \dots, j_n\}$ повинна бути призначена на виконання без переривань на одному приладі, який може працювати не більш, ніж з одним завданням одночасно. Прилад і завдання передбачаються безупинно доступними з моменту часу нуль, а простої приладу не допускаються. Завдання j , де $j = 1, 2, \dots, n$, вимагає часу виконання p_j і в ідеалі повинно бути закінчене у свій директивний строк d_j . Для окремих завдань задано час налагодження s_j , це означає, що в розкладі, у якому завдання виконується відразу після завдання i , повинен бути час налагодження s_j одиниць часу між моментом завершення завдання i , позначеним через C_i , і часом початку завдання j , що дорівнює $C_j - p_j$. Протягом цього періоду налагодження ніяке інше завдання не може бути виконано приладом. Часи налагодження є залежними від послідовності, тому що вони залежать як від i , так і від j . Для будь-якого заданого розкладу випередження та запізнення завдання j можуть бути визначені виразами (2) і (3). Мета полягає у тому, щоб знайти розклад, який мінімізує сумарне випередження і запізнення всіх завдань:

$$\sum_{j=1}^n (E_j + T_j), \quad (1)$$

де випередження і запізнення визначаються відповідно як:

$$E_j = \max(0, d_j - C_j) = (d_j - C_j)^+, \quad (2)$$

$$T_j = \max(0, C_j - d_j) = (C_j - d_j)^+. \quad (3)$$

2. Огляд існуючих результатів

Сформульована задача (позначимо її МВЗН) узагальнює задачу МВЗ [1, 2] і відноситься до класу NP -повних.

Складанню розкладів відносно директивних строків приділяється значна увага в літературі. Одна з причин цього явища – зростаючий тиск конкуренції на міжнародних ринках: фірми повинні запропонувати велику розмаїтість різних та індивідуальних виробів, у той час як клієнти очікують, що замовлені товари будуть поставлені вчасно. Принцип виробництва «точно в строк» установлює, що необхідна кількість товарів повинна бути вироблена або поставлена точно в заданий час. Виконання роботи з випередженням приводить до витрат на складування, у той час як запізнення робіт – до штрафів і, в остаточному підсумку, втраті доброзичливості клієнтів і репутації фірми.

У [4] розглядається актуальність задачі та приводиться огляд літератури зі складання розкладів «точно в строк» і розкладів з налагодженнями. Представлено алгоритм гілок і границь та евристична процедура одержання локального оптимуму. В [5] описується процедура змішаного цілочисельного програмування, яка дозволяє розв'язувати задачі невеликої розмірності.

На жаль, відомо замало публікацій про розв'язання цієї задачі, і точні методи на основі гілок і границь дозволяють розв'язувати задачі лише невеликої розмірності. Тому використовуються алгоритми локального пошуку як практичний

підхід для розв'язання задач комбінаторної оптимізації. Починаючи із припустимого розв'язку, ці алгоритми ітеративно пробують поліпшити поточний розв'язок, вишукуючи кращий розв'язок в околиці поточного розв'язку, поки не знайдений локальний оптимум. Ефективність цих алгоритмів якісно залежить від визначення околиці: при більших околицях якість розв'язку краще, але час обчислень більше. Тому на практиці більші околиці не корисні, якщо тільки вони не можуть ефективно досліджуватися.

3. Алгоритм розв'язання задачі

Евристичний алгоритм A розв'язання задачі МВЗН розроблено в [3].

Алгоритм складається з двох етапів.

На першому етапі налагодження не враховується. У блоці 1 розв'язується задача мінімізації сумарного запізнення при виконанні незалежних завдань одним приладом МСЗ [2]. Алгоритм побудовано на перестановках, він полягає в оптимальному використанні завдань, що запізняються, та резервів завдань, що не запізняються. Таким чином, реалізується зменшення сумарного запізнення за рахунок зменшення сумарного випередження. У блоці 2 здійснюється зменшення значення сумарного випередження і запізнення за допомогою послідовного збільшення моментів початку виконання завдань.

Позначимо $r_{\min} = \min\{r_j\}$, N_r – число завдань з резервами ($r_j > 0$); N_3 – число завдань, що запізняються (до їх числа включаються завдання з нульовим резервом).

Твердження [6] (використання резервів завдань, що випереджають). Якщо у послідовності σ виконується $N_r > N_3$, то при збільшенні початку виконання завдань на величину, рівну r_{\min} , значення функціонала випередження/запізнення зменшується на величину $(N_r - N_3)r_{\min}$.

Розглядається послідовність, що отримана в результаті розв'язання задачі МСЗ. Якщо в отриманій послідовності резерви відсутні, то ця послідовність оптимальна за критерієм МВЗ. На кожній ітерації за умови $N_r \geq N_3$ збільшуються моменти початку виконання завдань у поточній послідовності на r_{\min} . Такі процедури виконуються, поки не буде виконано умову $N_r < N_3$. Отриману послідовність позначаємо σ^R .

На другому етапі здійснюється оптимізація послідовності σ^R з урахуванням налагоджень приладу, що залежать від послідовності, за допомогою процедури локального пошуку ефективного розв'язання поставленої задачі. Аналізується послідовність. Для тих завдань, для яких задані налагодження приладу, включаємо ці налагодження у тривалості виконання завдань. Отриману послідовність позначаємо σ^{R1} .

Використовуються такі типи перестановок: API (суміжна попарна перестановка), NAPI (несуміжна попарна перестановка), EBSR (витяг і повторна вставка зі зрушенням назад) та EFSR (витяг і повторна вставка зі зрушенням уперед).

Нижче приводяться результати досліджень ефективності алгоритму A .

4. Обчислювальні результати

Алгоритм A був закодований мовою C# у середовищі розробки Visual Studio 2010 під бібліотеку Microsoft .NET 4.0. Випробування проводилися на персональному комп'ютері із процесором Pentium CORE 2 Duo 2.0 ГГц із оперативною пам'яттю 2 Гбайта під управлінням ОС Microsoft Windows Vista. Досліджувалися задачі розмірності до 250 завдань.

Для визначення ефективності алгоритму були проведені дослідження залежності часу розв'язання задачі від розмірності та від складності задачі, а також дослідження проценту відхілення розв'язку від оптимального.

Схема генерації даних, запропонована Фішером [7], використовувалася для тестування алгоритму на різних типах прикладів, тип задачі визначається комбінацією фактора запізнення T і діапазону директивних строків R . Для кожної задачі спочатку генеруються тривалості виконання і часи налагоджень з рівномірного розподілу із заданими границями. Потім обчислюються директивні строки з розподілу, рівномірного на

$$[p^*(1-T-R/2), p^*(1-T+R/2)],$$

де p^* – сума всіх тривалостей. Значення T і R вибираються з множин $\{0,2; 0,4; 0,6; 0,8\}$ і $\{0,2; 0,4; 0,6; 0,8; 1,0\}$, відповідно, даючи по 20 задач кожного типу.

5. Дослідження впливу значень факторів R і T

Для визначення впливу діапазону директивних строків R і фактора запізнення T була промодельована робота алгоритму на задачах з параметрами R і T у діапазоні від 0 до 1 з кроком 0,2. Для кожної пари значень критеріїв і її розмірності було розв'язано не менш, ніж 20 задач. Це дозволило визначити такі параметри i , при яких алгоритм буде розв'язувати задачі відносно швидко, і які значення параметрів є найбільш складними для досліджуваного алгоритму.

Щоб продемонструвати загальну картину на невеликих задачах, у табл. 1–4 представлено середньоарифметичний час розв'язання задач різної розмірності. З ростом розмірності характеристики будуть такими ж. Тривалості виконання обрані в межах [50, 100]. Кількість завдань, що потребують налагодження, обрана на межі 50%.

Таблиця 1

Середній час розв'язання задач для $n = 10$ (мс)

$T \backslash R$	0	0,2	0,4	0,6	0,8	1
0	0,1	0,1	0,1	0,1	0,1	0,1
0,2	0,2	0,6	0,3	0,1	0,1	0,2
0,4	0,2	22,1	10,9	1,8	0,4	0,4
0,6	0,2	26,3	44,2	47,0	63,3	76,9
0,8	0,2	2,1	2,0	5,1	23,3	68,2
1	0,2	0,4	0,5	0,6	1,1	3,8

Таблиця 2

Середній час розв'язання задач для $n = 15$ (мс)

$T \backslash R$	0	0,2	0,4	0,6	0,8	1
0	0,2	0,2	0,2	0,2	0,2	0,2
0,2	0,3	0,9	0,4	0,2	0,2	0,3
0,4	0,3	33,2	16,4	2,8	0,6	0,5
0,6	0,3	39,4	66,5	70,8	95,5	116,2
0,8	0,3	3,2	3,0	7,6	35,0	102,5
1	0,3	0,6	0,8	0,9	1,7	5,8

Таблиця 3

Середній час розв'язання задач для $n = 20$ (мс)

$T \backslash R$	0	0,2	0,4	0,6	0,8	1
0	0,2	0,2	0,2	0,3	0,3	0,3
0,2	0,4	1,2	0,6	0,3	0,3	0,4
0,4	0,4	44,2	21,9	3,7	0,9	0,7
0,6	0,4	52,5	88,8	94,8	127,7	155,6
0,8	0,4	4,2	4,1	10,2	46,6	136,9
1	0,4	0,8	1,1	1,2	2,3	7,8

Таблиця 4

Середній час розв'язання задач для $n = 25$ (мс)

$T \backslash R$	0	0,2	0,4	0,6	0,8	1
0	0,3	0,3	0,3	0,4	0,4	0,4
0,2	0,5	1,4	0,7	0,4	0,4	0,4
0,4	0,5	55,3	27,4	4,6	1,1	0,9
0,6	0,5	65,7	111,2	118,8	160,1	195,3
0,8	0,5	5,3	5,1	12,7	58,3	171,4
1	0,5	1,0	1,4	1,4	2,8	9,8

З результатів добре видно, що множина розв'язків задач із різними значеннями критеріїв i має точку максимуму зі значеннями $R=1,0$; $T=0,6$. Також можна виділити області з локальним максимумом у точках зі значеннями параметрів $R=0,2$; $T=0,6$ і $R=0,8$; $T=0,6$. Це саме ті області, у яких розв'язання задач є самим важким для алгоритмів. Зі збільшенням розмірності задач різниця у швидкодії між задачами з параметрами, віднесеними до області максимуму, і всіма іншими задачами зростає. Це пояснюється тим, що у більшості областей алгоритм працює поліноміальний час, у той час як у вищезгаданих максимумах він зростає експоненційно, відповідно різниця збільшується за експоненційним законом.

Утворюємо на основі таблиць 1–4 три комбінації параметрів R і T , що відображають для кожної розмірності найбільш складні випадки задачі – комбінація I: $R=1,0$; $T=0,6$; комбінація II: $R=0,8$; $T=0,6$; комбінація III: $R=0,2$; $T=0,6$.

6. Результати порівняння із методом гілок і границь

Для того щоб показати, які результати дає алгоритм на задачах різної розмірності для комбінацій I, II, III параметрів T і R , зведемо результати з таблиць 1–4 в табл. 5. У колонці «Опт» показано час оптимального розв'язання задачі, отриманого алгоритмом гілок і границь.

Таблиця 5

Середній час розв'язання t_{cp} у порівнянні з точним алгоритмом (с)

Параметри задачі		t_{cp}		Параметри задачі		t_{cp}	
n	Комбінація параметрів	Опт	A	n	Комбінація параметрів	Опт	A
10	I	3,2	0,077	20	I	40,5	0,156
10	II	3,3	0,063	20	II	38,2	0,128
10	III	3,1	0,026	20	III	31,0	0,053
15	I	15,9	0,116	25	I	115,9	0,195
15	II	16,2	0,095	25	II	119,1	0,160
15	III	14,8	0,039	25	III	77,2	0,066

Результати з табл. 5 показують, що алгоритм A був дуже ефективним. У середньому на розв'язання треба було менш, ніж 0,2 секунди для будь-якого набору з 20 задач. Середня кількість часу, необхідного на оптимальне розв'язання алгоритмом гілок і границь, становила не більше 2 хвилин. Однак, час, необхідний для оптимального розв'язання, швидко збільшувався при зростанні розмірності. Це вказує на те, що використовувати алгоритм гілок і границь для розв'язання задач більших розмірностей непрактично.

7. Вплив розмірності задачі на її розв'язання

У табл. 6 показана залежність середнього часу розв'язання від розмірності для найскладнішої комбінації параметрів I ($R=1,0$; $T=0,6$).

Таблиця 6

Залежність часу розв'язання t_{cp} від розмірності при $R=1,0$; $T=0,6$ (с)

n	Кількість завдань, що потребують налагодження, %									
	10	20	30	40	50	60	70	80	90	100
50	0,37	0,37	0,38	0,38	0,38	0,39	0,39	0,39	0,41	0,41
75	1,19	1,21	1,22	1,23	1,24	1,26	1,28	1,29	1,31	1,32
100	2,10	2,12	2,13	2,16	2,18	2,19	2,22	2,24	2,26	2,29
125	2,70	2,73	2,74	2,75	2,77	2,80	2,81	2,83	2,85	2,88
150	3,11	3,13	3,17	3,20	3,24	3,27	3,32	3,35	3,40	3,45
175	3,78	3,83	3,89	3,94	4,01	4,08	4,15	4,23	4,30	4,38
200	3,92	4,05	4,19	4,34	4,49	4,65	4,83	5,00	5,20	5,38
225	5,15	5,25	5,38	5,51	5,65	5,79	5,94	6,10	6,26	6,44
250	5,52	5,70	5,89	6,09	6,31	6,52	6,76	7,01	7,26	7,53

8. Ефективність алгоритму

У табл. 7 наведено середнє відхилення значення функціонала від оптимального у відсотках. Для алгоритму A відсоток відхилення Δ обчислюється таким чином:

$$\Delta = (Z - Z_{opt}) \cdot 100 / Z_{opt},$$

де Z_{opt} – сумарне випередження/запізнення розв'язку, виконаного алгоритмом гілок і границь, а Z – сумарне випередження/запізнення розв'язку, виконаного евристичним алгоритмом.

Таблиця 7

Середнє відхилення значення функціонала від оптимального, %

n	Комбінація параметрів		
	I	II	III
10	0,0	0,0	0,0
15	4,1	5,0	2,5
20	8,9	7,4	12,3
25	21,3	20,4	18,1

Результати з табл. 7 показують, що алгоритм гілок і границь приводить до кращого значення функціонала, ніж алгоритм A . Фактично, для усіх задач з розмірністю 10 алгоритм A знайшов оптимальний розв'язок. Середній відсоток відхилення для алгоритму A становив біля 8,3% на всіх множинах задач і менше 9% для 8 з 12 множин задач.

Висновки

Ми представили результати досліджень ефективності алгоритму розв'язання задачі МВЗН із [3]. Дослідження показали, що алгоритм A дає розв'язки, близькі до оптимальних, та дозволяє за прийнятний час ефективно розв'язувати задачі великої розмірності (табл. 6 і 7).

Список літератури: 1. Павлов, О.А. Дослідження властивостей та розв'язання задачі «Мінімізація сумарного штрафу як за випередження, так і за запізнення відносно директивних строків при виконанні незалежних завдань одним приладом» [Текст] / О.А. Павлов, О.Б. Місюра, О.В. Мельников // Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка: Зб. наук. пр. – К. : «ВЕК+», 2008. – №48. – С. 3–6. 2. Згуровский, М.З. Принятие решений в сетевых системах с ограниченными ресурсами [Текст] / М.З. Згуровский, А.А. Павлов : Монография.

– К. : Наукова думка, 2010. – 573 с. 3. Ващук, Ф.Г. Складання розкладів сумарного випередження і запізнення із налагодженнями, що залежать від послідовності [Текст] / Ф.Г. Ващук, О.А. Павлов, О.Б. Місюра, О.О. Мельник // Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка: Зб. наук. пр. – К. : «ВЕК+», 2011. – №53. – С.192–194. 4. Sourd F. Earliness–tardiness scheduling with setup considerations / F. Sourd // Computers and Operations Research. – 2005. – Vol. 32. – P. 1849–1865. 5. Coleman B.J. A simple model for optimizing the single machine early/tardy problem with sequence dependent setups / B.J. Coleman // Production and Operations Management. – 1992. – Vol. 1. – P. 225–228. 6. Згуровский, М.З. ПДС-алгоритмы и труднорешаемые задачи комбинаторной оптимизации [Текст] / М.З. Згуровский, А.А. Павлов, Е.Б. Мисюра // Системні дослідження та інформаційні технології. – 2009. – №4. – С. 14–31. 7. Fisher M.L. A dual algorithm for the one-machine scheduling problem / Fisher M.L. // Math. Programming. – 1976. – №11. – P. 229–251.

Надійшла до редколегії 7.12.2011

УДК 519.854.2

Исследование эффективности алгоритма составления расписаний суммарного опережения и запаздывания с наладками, зависящими от последовательности / Ф.Г. Ващук, Е.А. Мельник // Бионика интеллекта: науч.-техн. журнал. – 2012. – № 1 (78). – С. 49-52.

В статье рассмотрены результаты исследования эффективности алгоритма поиска локального оптимального решения задачи составления расписаний по критерию минимизации суммарного опережения и запаздывания. Приведены результаты сравнения указанного алгоритма с методом ветвей и границ для задач большой размерности. Исследования показали, что рассмотренный эвристический алгоритм дает решения, близкие к оптимальным.

Табл. 7. Библиогр.: 7 назв.

UDK 519.854.2

Research of efficiency of algorithm machine scheduling of minimizing the total earliness and tardiness that depend on sequence / Vashchuk F.G., Melnyk O.O. // Bionics of Intelligense: Sci. Mag. – 2012. – № 1 (78). – P. 49-52.

The results of the research of efficiency of algorithm to search a local optimal solution of the machine scheduling of minimizing the total earliness and tardiness problem are under consideration in this article. The results of comparison of the proposed algorithm with the branch and bound algorithm for large dimension problems are given. The researches have shown that the proposed heuristic algorithm gives the solutions close to optimum.

Tab. 7. Ref.: 7 items.